

Trustworthy and Efficient Routing Algorithm for IoT-FinTech Applications Using Non-Linear Lévy Brownian Generalized Normal Distribution Optimization

Ali Safaa Sadiq, *Senior Member, IEEE*, Amin Abdollahi Dehkordi, Seyedali Mirjalili, *Senior Member, IEEE*, Jingwei Too, Prashant Pillai, *Senior Member, IEEE*

Abstract—The huge advancement in the field of communication has pushed the innovation pace towards a new concept in the context of Internet of Things (IoT) named IoT for Financial Technology applications (IoT-FinTech). The main intention is to leverage the businesses' income and reducing cost by facilitating the benefits enabled by IoT-FinTech technology. To do so, some of the challenging problems that mainly related to routing protocols in such highly dynamic, unreliable (due to mobility) and widely distributed network need to be carefully addressed. This paper therefore focuses on developing a new trustworthy and efficient routing mechanism to be used in routing data traffic over IoT-FinTech mobile networks. A new Non-linear Lévy Brownian Generalized Normal Distribution Optimization (NLBGNDO) algorithm is proposed to solve the problem of finding an optimal path from source to destination sensor nodes to be used in forwarding FinTech's related data. We also propose an objective function to be used in maintaining trustworthiness of the selected relay-node candidates by introducing a trust-based friendship mechanism to be measured and applied during each selection process. The formulated model also considering node's residual energy, experienced response time, and inter-node distance (to figure out density/sparsity ratio of sensor nodes). Results demonstrate that our proposed mechanism could maintain very wise and efficient decisions over the selection period in comparison with other methods.

Index Terms—IoT, Optimization, FinTech, Trustworthy AI

I. INTRODUCTION

Internet of Things (IoT) has gained a lot of interest since the start of the last decade, when Kevin Ashton in 1999 has given the birth of this terminology [1]. Interestingly enough that his main intention behind inventing this term was to present the interesting benefits of using Radio-Frequency Identification

(RFID) in optimizing supply chain at his place of work, (Procter&Gamble). Since back then, the technology context of connected things that introduced by IoT has emerged and gained popularity, particularly at the summer of 2010 [1]. This new technology has changed the direction of addressing mechanism in Internet 360 degrees. Rather than addressing only Internet infrastructure, direction has dramatically changed to address our physical world, that including smart home security systems, autonomous farming equipment, wearable health monitors, smart factory equipment, and many more.

This big move towards addressing many smart connected things in our life has brought a crucial need for better strategy or method that could be used in driving the massive amount of generated data. This concept has stressed more on the term Big-Data that was credited first by John R. Mashey [2], as there will be huge amount of data associated with this incredible number of connected things around the globe within smart cities. Therefore, having the most efficient and effective method of routing this huge amount of data is daunting.

On the other hand, IoT context has been reformed into few different shapes during the last few years. These shapes were defined according to the useability of this technology in different domains. Some of these domains are, but not limited to agriculture, smart cities, smart/automated connected vehicles [3, 4], smart industry 4.0 and Financial Technology (FinTech). It is important to highlight that this paper will focus only on solving one of the common problems related to the IoT for FinTech as a way to show case our proposed method.

Recently, IoT technology has contributed a lot to different applications related to FinTech, such as supply-chain for a wide range of businesses [5]. Though, there is still a room of challenges need to be carefully addressed before we can fully rely on such highly automated technology. For instance, limited power resources been always the enemy of the smart connected things, this is due to the fact that most of these sensors or actuators are occupied with limited batteries [6]. For this reason, the Low Power and Lossy area networks (RPL)/IPv6 RPL protocol has been introduced at first by the Internet

AS Sadiq and P Pillai are with Wolverhampton Cyber Research Institute, School of Mathematics and Computer Science, University of Wolverhampton, Wolverhampton, WV1 1LY, UK. (Ali.Sadiq@wlv.ac.uk) (P.Pillai@wlv.ac.uk)
 AA Dehkordi with Computer Engineering Faculty, Najafabad Branch, Islamic Azad University, Najafabad, Iran. (amin.abdollahi.dehkordi@gmail.com)

S Mirjalili is with the Yonsei Frontier Lab, Yonsei University, South Korea and Torrens University Australia (ali.mirjalili@gmail.com)
 J Too is with the Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia. (jamesjames868@gmail.com)

Engineering Task Force (IETF) [7]. The main intention was to obtain the best route between nodes for passing data's traffic with the use of an objective function that looking for the shortest distance and least power consumption associated to the selected link. Even though there are many other dynamic factors associated with the intermediate nodes involving in forming the connection link, RPL needs more careful considerations to improve its performance. Such inaccurate decision will eventually lead for transmitting data over a link that consists of one or more of incompetent nodes, which will result in frequent retransmissions. Therefore, there were several studies in the literature have tried to overcome such issue with different attempts [8-10].

Security is one of the other inherent challenge with such lightweight routing protocols within IoT networks. Bear in mind these protocols are forming the underlyingly data transmission mechanism within the IoT for FinTech that introduce crucial security challenge [11, 12]. Inherently, these types of protocols require simple security solutions that could save energy and avoid complex procedures [13, 14]. Therefore, the wise selection of connection link within FinTech is really challenging as any successful attack would cause the loss of huge amount of money, considering the fact that the projection of having \$3.9 trillion to \$11.1 trillion per year in 2025 adding value to the global economy over IoT for FinTech technologies [15].

Therefore, and from the above discussion, proposing an optimal solution that can consider multifactor mechanism while selecting the best link for transmitting data over IoT that could be used for FinTech is essential [16]. Meta-heuristics algorithms are considered as one of the promising solutions for solving a wide range of complex optimization problems [17-19]. Though, meta-heuristics algorithms have own challenges that are trapping with the local and global search rejoins, which mainly occur with optimization problems with high dimensions. These challenges and proposed solutions in the state of the arts are discussed in the following section. Hence, in this paper, we will explore and exploit for the first time the use of our proposed non-linear Levy flight Generalized Normal Distribution Optimization (LGND) algorithm in forming an energy efficient and secure routing algorithm for FinTech applications based on IoT network.

The rest of the paper is organized as section II presenting the related work, while section III detailing out the proposed NLBGND algorithm, section V demonstrating the problem formulation of IoT-FinTech along with results and analysis. The paper was finally concluded by section VI.

II. RELATED WORK

IoT technology has introduced a wide range of applications that could be offering different types of services. These applications could be covering healthcare, smart cities, smart industries, FinTech and many more. In the literature, there was a huge focus on the use of Wireless Sensor Networks (WSNs) in facilitating the underlaying infrastructure of IoT technology for most of this wide spectrum of applications. While for routing the data traffic of these applications, RPL protocol has

gained the biggest chunk of the research focus, by developing different approaches to offer the main key requirements of IoT's applications. Some of the main crucial factors that RPL or any other similar efficient routing protocol needs to offer as stated in [7], are energy efficiency, reliability, security, responsiveness, scalability, and mobility.

Considering the use of IoT for some of the supply-chain applications that supports FinTech, multiple sensor nodes need to be deployed; some of them fixed while others are mobile. Therefore, and from the nature of these sensors, they could face some technical issues as well as some other environmental factors that need to be considered while forming a link/route from source to destination. The authors in [20] have introduced a mobility aware version of RPL that could overcome some of the issues with the mobile context, which named GTM-RPL. In their method, game theory strategy was introduced to improve packet delivery ratio, energy efficiency and delay. Hence, their proposed method changes the transmission range according to the network condition. However, game theory-based decision-making process offers unneglectable amount of uncertainty and unreliability. This is due to its nature that relying mainly on the predefined set of rules and players that would highly reflect on the quality of the selected solutions [21-23].

On the other hand, another version of mobile RPL solution was introduced by [24]. The main intention was to minimize the time taken by each handoff process within mobile sensor nodes. The authors have introduced the use of four timers to the native trickle algorithm as a way to identify the disconnected sensor nodes. Though, their method was tested with a single mobile sensor node that moving with a fixed velocity crossing some other fixed deployed nodes. Additionally, there was no clear discussion on the objective function formulation and how it has been used in finding the optimal path from source to destination.

The native RPL protocol creates a Directed Acyclic Graph (DAG) without outgoing edges that represents the base element of the topology, this guarantee that no endless loop exists in the chain of nodes. Accordingly, the sink node commences constructing the initial DAG by setting itself as the best DAG root. While the other nodes in the RPL's graph begin creating their own DAGs, by calculating the route from themselves towards the root node. To build all these routes options across each interconnected node within DAG, RPL protocol employs several control messages. The default DAG Information Object (DIO) that sent by the root node consists of the rank of the relaying node, the node's ID, the version number, and the DAG-ID. Using this information, the node will formulate a decision for the route calculation.

To have more informative decision, a queue utilization scheme was introduced by adding congestion information from each DIO message to formulate load balancing approaches, which was developed by the authors in [25, 26]. These approaches were successful in achieving load balancing and improving to an extend the performance. Though, these types of protocols have changed the native RPL by introducing control messages, which have by nature changed the DAG formation procedure. This type of changes made on the protocol

messages' structure will not allow a scalable deployment of many sensors' nodes for the IoT for FinTech applications as they will not be adaptable with the native RPL, which inquires some modifications prior to the deployment.

Therefore, there should be a way to optimize the native RPL with more interoperability for IoT-FinTech applications to make it more flexible and scalable, to introduce it as the best choice for IoT routing. To achieve that, recently, there was a noticeable focus on applying some sort of metaheuristic algorithms and AI-based solutions to optimize the decision made by RPL routing algorithm. For instance, authors in [27], have proposed a congestion alleviation method, which uses grey theory. Their proposed method considers buffer occupancy, expected transmission count and queuing delay in a multi objective optimization algorithm. On the other hand, in [28] and [29] authors proposed new routing algorithm for IoT networks that using reinforcement learning and fuzzy logic. In another attempt and in order to further improve the routing efficiency in IoT networks, the authors in [30] have proposed a heuristic method to obtain the optimal path from source to destination. The proposed method has considered the energy of transmitted data from sensor node to base station.

However, from the above-mentioned state of the arts, at some points, there are still some aspects need to be wisely considered, such as how much trust we should give on the intermediate nodes that normally been selected in forwarding and routing formation? How can we quantify security and trust to be considered in the selection process? Some of the used optimization or AI algorithms are associated with a degree of uncertainty or stagnation within local optimal solution, which eventually make them loss the opportunity to explore more promising regions in finding better solutions that could effectively improve the QoS of the applications running on the IoT for FinTech.

Hence, the main contributions of this paper listed as:

- Boosting the performance of the native GND0 algorithm, the main contribution of the proposed method is introducing two new strategies (Lévy flight and Brownian motions) to enhance tuning the exploration and exploitation phases of GND0 algorithm's controlling parameters.
- Introducing for the first time the use of the new developed Non-Linear Leavy Flight Generalized Normal Distribution Optimization NLBGND0 algorithm in solving formulated optimization routing problem that considers trustworthy membership function.
- Implementing NLBGND0 algorithm for solving RPL's routing problems for supply-chain related problems in IoT for FinTech.

The next section III introduces the design detail of the proposed method that will be used in optimizing RPL protocol for IoT-FinTech applications. Section IV presents the numerical experiment, performance evaluation and discussion. While Section V demonstrating the implemented application for efficient and trustworthy routing algorithm for IoT-FinTech using NLBGND0. Finally, the paper was concluded

in Section VI.

III. NONLINEAR LÉVY BROWNIAN GENERALIZED NORMAL DISTRIBUTION OPTIMIZATION (NLBGND0)

This section details out the specifications of our proposed method. It begins with the presentation of all the preliminaries of the standard GND0 algorithm, and then follows with the details of the modified version of GND0, which formulates our proposed NLBGND0 optimization algorithm.

A. Inspiration of GND0

The GND0 algorithm is one the latest meta-heuristic algorithms that mainly introduced to solve problems within the field of optimization. This algorithm is inspired by a mathematical theory known as the normal distribution theory (Gaussian distribution). This theory is very important as it has been known as a suitable tool to describe natural phenomena. Assuming that the random variable x follows a probability distribution with the location parameter μ and the scale parameter δ and its probability density function. Theoretically, this distribution is defined as follows [31].

$$f(x) = \frac{1}{\sqrt{2\pi}\delta} \exp\left(-\frac{(x-\mu)^2}{2\delta^2}\right) \quad (1)$$

According to Eq. (1), x can be defined as a normal random variable and this distribution can also be as a normal distribution, which formulated as $x \sim N(\mu, \delta)$. The normal distribution consists of two variables: location parameter (μ) and scale parameter (δ). These parameters are used to express the mean and standard variance of the random variables, respectively.

Generally, GND0, like population-based optimization methods, consists of 3 steps in the searching process. First, all the primary search agents generated in a distribution are scattered. In the second stage, all search agents, under the guidance of the exploration and exploitation phases, move towards the global optimal solution. Finally, all the factors are gathered around the optimal solution will be obtained. In GND0, these 3 steps can be described by multiple normal distributions. In this algorithm, the positions of all agents are considered as random variables exposed to natural distribution. Due to the structure of GND0, in the first stage there is a large gap between the mean position and the optimal position. Hence, the standard variance of all agent positions is also very large. In the next step, as the position of the agents is updated, the distance between the mean and the optimal positions gradually decreases. Obviously, by reducing this distance at this stage, the

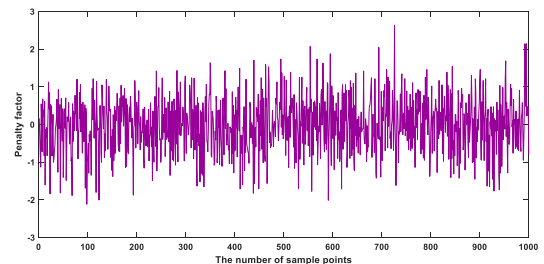


Fig. 1. The random sequence generated by Eq. (5).

standard variance of the position of all agents will gradually decrease. In the third step, the distance between the mean position and the optimal position, as well as the standard variance of the position of all search agents will be minimized.

B. The structure of GNDO

One of the advantages of GNDO is its very simple structure. The training strategy in GNDO is designed in two phases of local exploitation and global exploration. The local exploitation phase in this algorithm is based on the generalized normal distribution model, which is driven by the current mean position and the current optimal position [31]. The second phase is related to the random selection of 3 agents. In the following subsections, the two mentioned phases have been examined and discussed in more detail.

1) Exploitation

This phase is defined in optimization algorithms as the process of finding better solutions according to the current positions of all search agents around the search space. In GNDO, based on the relationship between the distribution of individuals in the population and the normal distribution, a generalized normal distribution model for optimization is defined as follows.

$$v_i^t = \mu_i + \delta_i \times \eta, i = 1, 2, 3, \dots, N \quad (2)$$

In Eq. (2), v_i^t is the trajectory vector of the factor at time t , μ_i is the generalized mean of the position i_{th} agent. δ_i is the generalized standard variance and η is the penalty factor. In addition, μ_i , δ_i and η are defined as follows.

$$\mu = \frac{1}{3}(x_i^t + x_{Best}^t + M) \quad (3)$$

$$\delta = \sqrt{\frac{1}{3}[(x_i^t - \mu)^2 + (x_{Best}^t - \mu)^2 + (M - \mu)^2]} \quad (4)$$

$$\eta = \begin{cases} \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2), & \text{if } a \leq b \\ \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2 + \pi), & \text{otherwise} \end{cases} \quad (5)$$

While a , b , λ_1 , λ_2 are random numbers in the scale of 0 to 1 in Eq. (3) x_{Best}^t is the best current position and M is the average position of the current population. In addition, M is calculated as follows.

$$M = \sum_{i=1}^N x_i^t \quad (6)$$

In GNDO, the three parameters used (e.g. μ, δ, η), which are playing a very important role in the exploitation strategy. The role of these parameters is discussed briefly below.

μ_i , is introduced as the generalized mean position in this algorithm, which is referring to the best current agent in x_{Best}^t that contains useful information about the optimal global solution [1]. According to the definition of μ_i , is for the i_{th} agent of x_i^t that is shifted to the best current agent of x_{Best}^t . This mechanism increases the likelihood of the i_{th} agent seeking for a better solution.

One of the drawbacks of this strategy in GNDO is the premature convergence and trapped in the local optima points. Due to the fact that the agents imitate the best agent in the population and step towards it, these agents are having higher

chances of being trapped into the local optimum position. Eventually, the entire population of the algorithm has no choice but to be trapped in a local optimum, which will eventually lead to a premature convergence. The solution proposed in GNDO to solve this problem is to specify the auxiliary parameter M as defined in Eq. (6). In this equation, M is the average position of the current population in the algorithm. Defining this parameter in GNDO increases the chance of the algorithm to escape from local optimal points.

The second influential parameter in GNDO is the generalized standard variance or δ_i . Purposing the use of this parameter in this algorithm is to improve the algorithm's local search capability. According to Eqs. (3, 4), this parameter can be considered as a random sequence to improve the local search around the average generalized position μ_i . A closer look at Eq. (6) and its role in the algorithm, when the distance between the position of the i_{th} agent of x_i^t , the mean position μ_i and the position of the best agent x_{Best}^t getting higher, the fluctuation of the random sequence produced will get stronger. In another words, using this random sequence helps those agents with high cost and little probability of finding a better solution around them to search for a better solution. In contrast, when agent x has a small cost with high probability of finding a better solution around itself, a random sequence with a weak fluctuation allows it to find a better solution.

In the GNDO algorithm, a penalty parameter called η is used to maximize the random generation of the generalized standard variance [1]. According to Eq. (5) and the random sequences generated in Fig. 1, it can be clearly seen that most of the random sequences are generated in intervals 1 and -1. In GNDO, the generalized standard variance is defined to always have a positive value. Therefore, using this punitive parameter can increase the search directions in the optimization problem space by GNDO, which improves and enhances the search capability of the algorithm.

2) Exploration

This phase is defined in GNDO to better explore the search space globally as a way to reach more promising areas. The exploration phase in GNDO is running based on three randomly selected factors, which are defined as follows:

$$v_i^t = x_i^t + \beta \times (|\lambda_3| \times v_1) + (1 - \beta) \times (|\lambda_4| \times v_2) \quad (7)$$

While λ_3 and λ_4 are two random numbers under the standard normal distribution. β as a regulating parameter with a random number between 0 and 1. v_1 and v_2 are two sequence vectors defined as follows.

$$v_1 = \begin{cases} x_i^t - x_{p1}^t, & \text{if } f(x_i^t) < f(x_{p1}^t) \\ x_{p1}^t - x_i^t, & \text{otherwise} \end{cases} \quad (8)$$

$$v_2 = \begin{cases} x_{p2}^t - x_{p3}^t, & \text{if } f(x_{p2}^t) < f(x_{p3}^t) \\ x_{p3}^t - x_{p2}^t, & \text{otherwise} \end{cases} \quad (9)$$

While $p1, p2, p3$ are three random numbers that are selected in the range of 1 to N , As such $p1 \neq p2 \neq p3 \neq i$. According to Eqs. (8, 9), the second part of Eq. (7) is called the term local learning. This means that agent i learns from agent $p1$, which is represented by v_1 . The third part of Eq. (7) is called the global

learning term, which represents the learning of agent i from agents $p2$ and $p3$. Adjustment parameter β is used to create a balance between these two parts. In addition, λ_3 and λ_4 are random numbers with standard normal distributions. These two parameters help GNDO to have more search space in the global search, which improves the process [31]. In GNDO, a screening mechanism is used to provide a better solution for the next generation of the population if the i_{th} agent failed in finding a better solution during the exploration and exploitation processes. This mechanism is described as follows.

$$x_i^{t+1} = \begin{cases} v_i^t, & \text{if } (v_i^t) < f(x_i^t) \\ x_i^t, & \text{otherwise} \end{cases} \quad (10)$$

C. Analysis of GNDO algorithm

Before explaining the proposed method, it is necessary to review its search pattern. Four benchmark functions with two decision variables are used in this section to evaluate the algorithm's performance. Fig. 2 shows how GNDO solves two unimodal problems with just one optimum and two multimodals with several optimal solutions.

This figure shows the form of the search space, the history of the sampled points, the trajectory of one solution, the average fitness of all solutions and the convergence curves. In the experiment, five solutions are used over 500 iterations.

Inspection of subplots in Fig. 2, it is clear that the history of the search for solutions was not presented in detail. Apart from finding a relatively poor estimate of the optimal in all test functions, it proves that the exploration should be more comprehensive, particularly for problems with many variables. The experiment is performed with two decision variables, and the exploration is still not as comprehensive as enough. This is going to be the first drawback that we are going to find the way to relieve. On the other hand, the trajectories, the average fitness (objective values), and the convergence curves indicate that after approximately two-thirds of iterations, the algorithm unexpectedly has switched to exploitation. This could lead to premature convergence. We argue that there should be a smooth transition from exploration to exploitation, which would be the second drawback that we will be looking for a remedy by the proposed method.

D. The proposed NLBGND method

The GNDO algorithm simulates the normal distribution theory, which is the Gaussian distribution. While having a reasonable structure and simple implementation, GNDO has very poor exploration and exploitation rates and it is suffering from being stuck in optimal local points, premature convergence and not achieving the global optimum. In this regard, an enhancement and improvement are required to boost the performance of the GNDO algorithm in optimizing more complex problems. The main motivation of proposed method is to enhance GNDO exploration and exploitation by modifying these twophases with two additional strategies: (i) Lévy flight and (ii) Brownian motions, and the algorithm's exploration and

exploitation phases using the control parameter. In the following sub-sections, we will briefly review the two mentioned strategies and discussing in detail the proposed method.

Lévy flight

This approach is a random walk created by the Lévy Flight Function. The performance of this function will monitor step sizes based on the probability function defined by the Lévy distribution (power-law tail) [2]:

$$L(x_i) \approx |x_i|^{1-\alpha} \quad (11)$$

When the length of the flight is described by x_i , the power level (power-law exponent) is demonstrated by $1 < \alpha \leq 2$. However, a fast and accurate algorithm has been proposed by Mantegna, which is used to generate more stable levy process. This method generates an arbitrary value of the index distribution (α) between 0.3 and 1.99. Due to these advantages, our proposed method will be using this strategy to produce random numbers based on the Lévy distribution as follows:

$$Levy(\alpha) = 0.05 \times \frac{x}{|y|^{1/\alpha}} \quad (12)$$

Where x and y are characterized by equations (13) and (14), two normal distribution variables with the standard deviations of σ_x and σ_y are defined as follows:

$$x = Normal(0, \sigma_x^2), \quad (13)$$

$$y = Normal(0, \sigma_y^2), \quad (14)$$

While the standard deviations are described in Eqs. (13, 14), σ_x and σ_y are defined as follows:

$$\sigma_x = \left[\frac{\Gamma(1 + \alpha) \sin(\frac{\pi\alpha}{2})}{\Gamma(\frac{(1 + \alpha)}{2}) \alpha 2^{\frac{(\alpha-1)}{2}}} \right]^{1/\alpha}, \sigma_y = 1, \alpha = 1.5 \quad (15)$$

Where α is generated in a range of 0.3 and 1.99. However, in this equation, its value is set to 1.5.

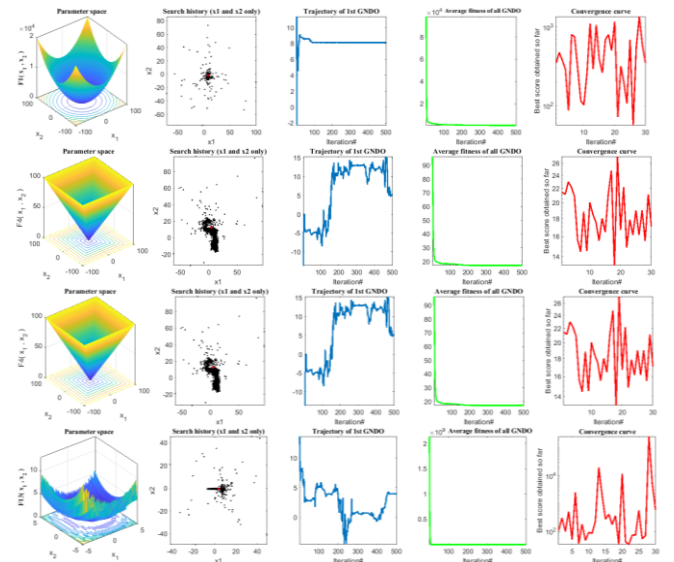


Fig. 2. Analysis of the GNDO's search behavior and performance when solving unimodal and multi-model test functions.

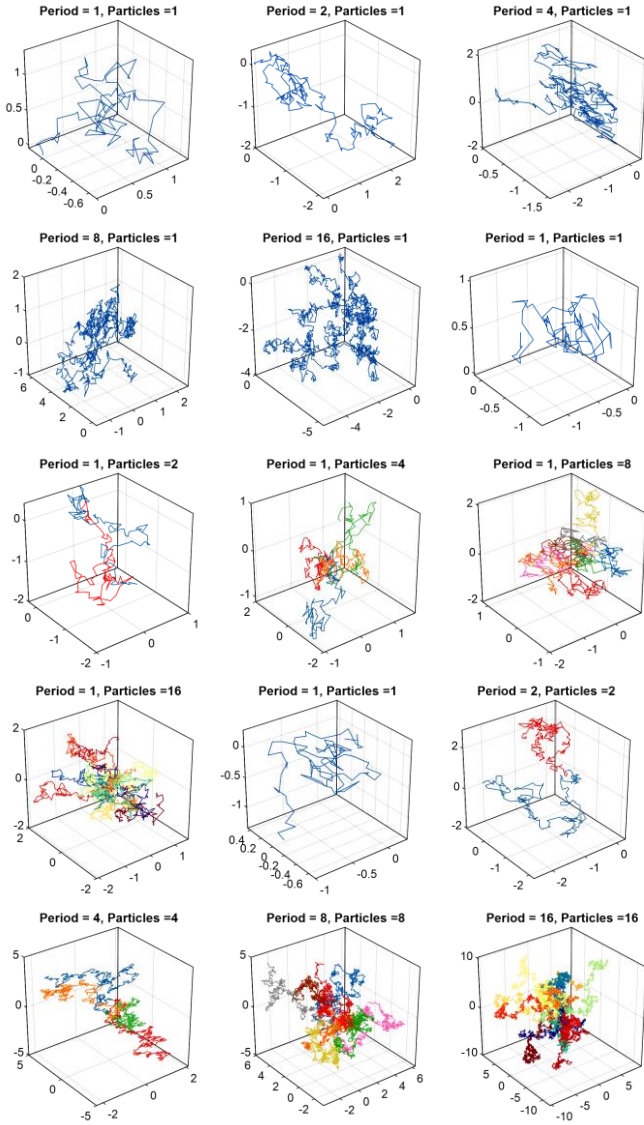


Fig. 3. Brownian walking randomly while changing the number of steps and the number of random walks.

1) Brownian motion

This motion strategy is the stochastic way in which the agent's phase length is determined by the probability function defined by the zero mean ($\mu = 0$) and the unit variance ($\sigma^2 = 1$) of the normal (Gaussian) distribution. The motion can be described at point x using the primary distribution function, the Probability Density Function (PDF) as follows:

$$f_B(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \quad (16)$$

Which σ^2 is the unit variance that its value is set to one, and μ is the zero mean that is specified in the normal (Gaussian) distribution. According to the definitions provided, Lévy's strategy typically tracks the area with small steps associated with long jumps. Because of these features, this strategy can provide accurate and efficient exploration of the surrounding neighborhood and other domain areas. In comparison to Lévy's

strategy, Brownian's motion concerns an environment with a standardized and regulated procedure. On the other hand, Brownian's motion has no capacity to search systematically and efficiently for different domain areas, the same as Lévy's strategy. It is important to recognize that using each of these methods on an individual basis does not yield positive results. Thus, the proposed algorithm has used a combination of these two techniques simultaneously to promote a standard exploration-exploitation-enabled framework. Due to the specific attributes of each strategy, the exploitation process is simulated by Lévy's strategy and Brownian motion has been used in the exploration phase of the algorithm.

2) Brownian Random Walk

One of the main inspirations of the proposed algorithm in this research is to add random walking behaviors produced by Brownian motion to GNDO. An example of a Brownian random walk is shown in Fig. 3. In this figure, each axis shows the steps and the essential control parameters, which are the period and the agents. The former parameter is the number of steps, and each unit is 100 steps. The number of particles is also the number of random walks.

As for the random walks in Fig. 3, if the number of steps is increased, the random walking path will be longer and more regions will be covered, as can be seen in the first two rows. The second row is where we produce twice as many random walks per time. It can be seen that each random walk is unique, and with plenty of them, a significant proportion of the space will be covered. Eventually, the last row is where we are increasing both the period and the number of agents. It is obvious that such mechanisms have provided a good exploration of the solution space. The figure indicates an example of a 3D space. Using such random walks to search around the solution space of an optimization problem is one of the main contributions of the proposed method to modify the exploration phase in GNDO.

3) The structure of proposed method

This section addresses the proposed algorithm structure called Nonlinear Levy Brownian Generalized Normal Distribution Optimization (NLBGND). The main motivation of the proposed NLBGND is to advance the efficiency of the original GNDO algorithm by two stages; the first is to combine the GNDO with the Levy-flight and Brownian motion, while the second stage is to use the non-linear control parameter as an adjustment variable between the exploration and exploitation phases. This combination in the basic GNDO uses the advantages of the levy flight and the Brownian motion theorem, which increases the algorithm's ability to explore and make more effective use of the search space. Moreover, the non-linear control parameter is used to increase GNDO's efficiency by adjusting the exploration and exploitation phases.

As stated in the previous sections, GNDO does not have acceptable exploration and exploitation phases. The main aim of the NLBGND algorithm is therefore to eliminate this apparent weakness in the GNDO algorithm. Due to the important role of η in GNDO in enhancing the ability of the algorithm to locally search for solutions of optimization problems in order to amplify the process of exploitation in this

algorithm. Hence, the effect of this parameter is improved by the use of Lévy Flight that introduced in the proposed method. Thus, Eq. (5) in the GNDO algorithm is redefined as follows.

$$\eta = \begin{cases} \vec{R}_L \otimes \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2), & \text{if } a \leq b \\ \vec{R}_L \otimes \sqrt{-\log(\lambda_1)} \times \cos(2\pi\lambda_2 + \pi), & \text{otherwise} \end{cases} \quad (17)$$

Where R_L generates random numbers based on Lévy distribution to simulate the Lévy movement.

Due to the remarkable characteristics of Brownian motion, this theory has been used in the proposed method to amplify the exploration phase of the GNDO algorithm. Hence, Eq. (7) in the native GNDO algorithm is modified in the proposed algorithm as follows.

$$v_i^t = c_p \times \vec{R}_B \otimes (x_i^t + \beta \times (|\lambda_3| \times v_1) + (1 - \beta) \times (|\lambda_4| \times v_2)) \quad (18)$$

Where R_B is a random number, generating based on the Normal distribution shows the Brownian motion. The c_p is a nonlinear control parameter defined for adjusting the balance between exploration and exploitation. This parameter is defined as follows:

$$c_p = e^{-\left(\frac{6t}{T}\right)^2} \quad (19)$$

Note that in Eq. (18), t is the current iteration and T is the total number of iterations.

Algorithm 1 Pseudo-code of NLBGND algorithm

Input: populations size N , the Upper bound u , the lower bound l , the current number of iteration $t=0$ and the maximum number of iteration T .

```

Initialize populations size  $N$ ,  $i=1, \dots, n$ 
Calculate the fitness and construct the optimal solution  $x_{Best}$ 
Update the current number of iteration  $t=t+1$ 
►Main loop
While  $t \leq T$  do
    for  $i = 1:N$ 
        Generate a random number  $\alpha$  in  $[0,1]$ 
        if  $\alpha > 0.5$ 
            ►Exploitation strategy
            Select the current optimal solution  $x_{Best}$ 
            Calculate the mean position  $M$  by Eq. (6)
            Compute generalized mean position  $\mu$ , generalized
            Standard variance  $\delta$ , and new penalty factor by Eq. (3). Eq. (4) and Eq. (17) respectively.
            Perform the exploitation phase by Eq. (2) and Eq. (10)
        Else
            ►Exploration strategy
            Perform the exploration phase by Eq. (18), Eq. (8), Eq. (9) and Eq. (10)
        End (if)
    End (for)
    Update the current number of iterations  $t=t+1$ 
End while

```

Output: the optimal solution x_{Best}

The level of exploration and exploitation is affected not only by the variety of possible solutions, but also by the different operators which control the search intensity at different stages. While the use of levy flight and Brownian motions will enhance

both, exploration and exploitation phases of the GNDO, which it is very important to maintain a balance between these two phases of the metaheuristic methods in order to achieve a reliable result. Consequently, in our modified version of the GNDO algorithm, a non-linear control parameter is used to align the phases of exploration and exploitation. It also ensures that the transition from exploration to exploitation in the proposed method is carried out smoothly.

The pseudo-code for the proposed NLBGND algorithm is specified in Algorithm 1, where the highlighted parts showing the modifications applied to the native GNDO's exploration and exploitation phases.

It is important to mention that the computational complexity of the NLBGND algorithm in theory is identical to the native GNDO algorithm as the nonlinear Lévy Brownian equations are used instead of constant values for the main controlling parameters. Hence, the achieved improvement has been obtained without introducing any additional computational cost for the algorithm.

IV. NUMERICAL EXPERIMENT, PERFORMANCE EVALUATION AND DISCUSSION

This section first describes the experimental methods used in this study, the performance evaluation and then discusses the results. Like with other studies on metaheuristics, the analysis approach is quantitative. The proposed algorithm is evaluated on a broad variety of test functions to be represented in the following sub-section. Due to the stochastic nature of NLBGND, each experiment has been carried out 30 times and descriptive statistical indicators are used to report the reliability of the results. The performance of the proposed algorithm is contrasted against a number of traditional and well-known recent optimization algorithms for efficiency and results evaluation as well as validation. Moreover, the scalability and sensitivity analysis are also examined in order to assess their impact on the results and performance of NLBGND.

A. Experiments details

Various tests are designed in this section to evaluate the efficacy of the proposed method as set out in the preceding sections. To determine the capacity of NLBGND to explore, exploit and escape from local minima, a well-known set of benchmark functions, that including 23 unimodal, multimodal and multimodal fixed-dimensional functions. Also, six composite functions, which are widely used to test the performance of optimization algorithms, have been selected based on literature.

Unimodal functions (F1-F7), which also have a single optimum solution, deliberately evaluate the algorithm's ability to exploit them (see Table 1). The next group contains multimodal functions (F8-F13) (Table 2) with even more than one optimal solution. Local optimal solutions decide the algorithm's exploration efficiency in these functions, although an algorithm has to be able to search a given solution space globally and prevent being stuck in a local optimum as a way to find a global optimum. Multimodal fixed-dimensional functions (F14-F23) are the other group described in Table 4,

identical to multimodal functions, but with low and fixed dimensions. Functions 24-29 comprise the Composite Benchmark functions proposed at the 2014 Special Session of the CEC. These functions have been constructed using shifted, rotated, extended, and combining the most complex types of mathematical optimization problems published in the literature. These functions, including with their dimensions used in this research, have been listed and detailed in Tables 1-4.

TABLE 1 UNIMODAL BENCHMARK FUNCTIONS

Functions	Dimensions	Range	f_{min}
$f_1(x) = \sum_{i=1}^n x_i^2$	30,100,500,1000	[-100,100]	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30,100,500,1000	[-10,10]	0
$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30,100,500,1000	[-100,100]	0
$f_4(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	30,100,500,1000	[-100,100]	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2]$	30,100,500,1000	[-30,30]	0
$f_6(x) = \sum_{i=1}^n ((x_i + 0.5))^2$	30,100,500,1000	[-100,100]	0
$f_7(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0,1]$	30,100,500,1000	[-1.28,1.28]	0

TABLE 2 MULTIMODAL BENCHMARK FUNCTIONS [10]–[12].

Functions	Dimensions	Range	f_{min}
$f_8(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	30,100,500,1000	[-500,500]	-418.9829 > n
$f_9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30,100,500,1000	[-5.12,5.12]	0
$f_{10}(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	30,100,500,1000	[-32,32]	0
$f_{11}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	30,100,500,1000	[-600,600]	0
$f_{12}(x) = \frac{\pi}{n} (10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + \sin^2(\pi x_{i+1})] + (y_n - 1)^2) + \sum_{i=1}^n u(x_i, 10, 100, 4)$	30,100,500,1000	[-50,50]	0
$y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$			
$f_{13}(x) = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30,100,500,1000	[-50,50]	0

TABLE 3 PROPERTIES AND SUMMARY OF CEC-BC-2014 test functions.

Functions	Dimensions	Range	f_{min}
CF24 = Composition Function of Schwefel's Function, Rotated Rastrigin's Function and Rotated HGBat Function	30	[-100,100]	2400
CF25 = Composition Function of Rotated Schwefel's Function, Rotated Rastrigin's Function and Rotated High Conditioned Elliptic Function	30	[-100,100]	2500
CF26 = Composition Function of Rotated Schwefel's Function, Rotated HappyCat Function, Rotated High Conditioned Elliptic Function, Rotated Weierstrass Function and Rotated Griewank's Function	30	[-100,100]	2600
CF27 = Composition Function of Rotated HGBat Function, Rotated Rastrigin's Function, Rotated Schwefel's Function, Rotated Weierstrass Function and Rotated High Conditioned Elliptic Function	30	[-100,100]	2700
CF28 = Composition Function of Rotated Expanded Griewank's plus Rosenbrock's Function, Rotated HappyCat Function, Rotated Schwefel's Function, Rotated Expanded Scaffer's and Rotated High Conditioned Elliptic Function	30	[-100,100]	2800
CF29 = Composition Function of Hybrid Function 1, Hybrid Function 2, Hybrid Function 3	30	[-100,100]	2900

TABLE 4 FIXED-DIMENSION MULTIMODAL BENCHMARK FUNCTIONS [10]–[12].

Functions	Dimensions	Range	f_{min}
$f_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{\sum_{i=1}^2 (x_i - a_{ij})^6} \right)^{-1}$	2	[-65,65]	1
$f_{15}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_i(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	4	[-5,5]	0.00030
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	[-5,5]	-1.0316
$f_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos x_1$	2	[-5,5]	0.398
$f_{18}(x) = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_2^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_2^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2,2]	3
$f_{19}(x) = -\sum_{i=1}^4 c_i \exp \left(-\sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2 \right)$	3	[1,3]	-3.86
$f_{20}(x) = -\sum_{i=1}^4 c_i \exp \left(-\sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right)$	6	[0,1]	-3.32
$f_{21}(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	10.1532

$f_{22}(x) = -\sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	10.4028
$f_{23}(x) = -\sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	10.5363

B. Experimental Setup

We have evaluated the efficiency of the proposed NLBGND against some well-considered algorithms and recent ones described in the literature, including DE, PSO, WOA, SSA, MVO, MFO, GWO and the original GND, in order to verify the effectiveness and compare the results. The parameters setup for traditional algorithms are listed in Table 5. It is also important mentioning that the proposed method and other benchmarked algorithms have been implemented in Matlab R2018a built on a Windows 8.1 64-bit and 6 GB RAM computer.

TABLE 5. THE ALGORITHMS PARAMETER SETTINGS.

Algorithm	Parameter	Value
PSO	Topology fully connected	Inertia factor
	c_1	2
	c_2	2
DE	Scaling factor	0.5
	Crossover probability	0.2
MVO	maximum of Wormhole Existence Probability	1
	minimum of Wormhole Existence Probability	0.2
MFO	Convergence constant a	[-1 -2]
SSA	c_1 (balancing parameter of exploration and exploitation)	[1 0]
GWO	Convergence constant a	[2 0]
WOA	Convergence constant a	[2 0]

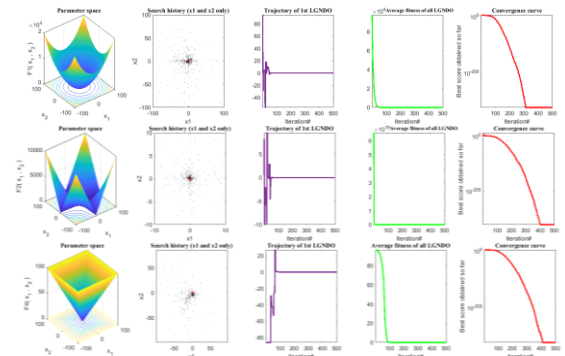


Fig. 4. Qualitative results for unimodal F1, F2, and F4 benchmark functions.

The definition of the parameters were based on the parameters that originally used by each algorithm's developer or the parameters that commonly used by the many researchers in the state of the art. In addition, all population and iterations of the optimization algorithms are set at 30 and 500, respectively.

C. Qualitative results

The results of the qualitative analysis of the NLBGND for solving unimodal and multimodal functions are discussed and evaluated in this section. Also, the qualitative analysis conceptually assesses positional changes as well as changes in fitness during the optimization process. Figs. 4-6 displaying four well-known indicators: search history, the trajectory of the first agent, average fitness, and convergence curve. The trajectory subplot monitors how the first agent variable shifts in the optimization process. The average fitness curve indicates

improvements in the average fitness of the population during optimization. Across the iteration process, the convergence curve shows that NLBGND has reached an optimum fitness value with a reasonably smooth convergence towards the end of the iteration process.

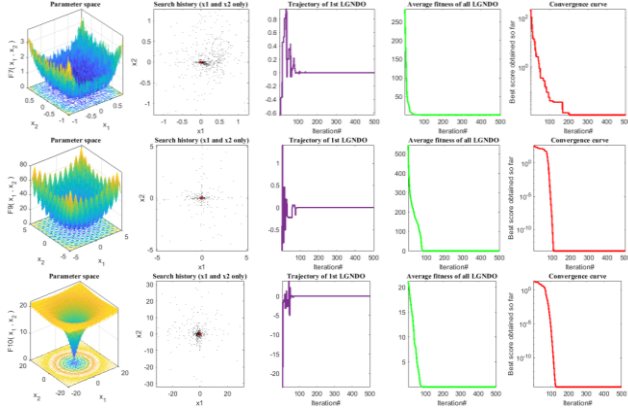


Fig. 5. Qualitative results for F7, F9, and F10 benchmark functions.

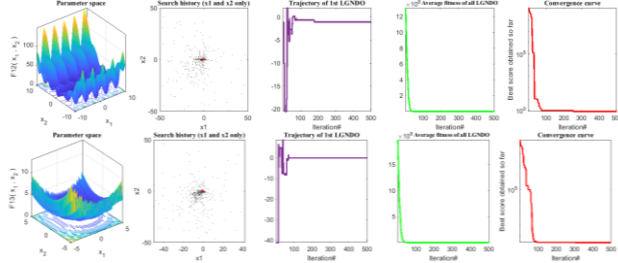


Fig. 6. Qualitative results for F12 and F13 benchmark functions.

It can be obviously seen with the subplot of the search history in Figs. 4–6 that NLBGND versus to GNDO has a strong coverage of the search space when concentrating on promising regions. The trajectory figure reveals that the solutions in NLBGND were confronted with abrupt changes in the initial stages, which eventually collapsed in the final stages. This virtually guarantees that the NLBGND eventually converges to a position and exploits the region of interest. The trajectories in the figs. 4–6 demonstrate the primary exploratory activity of the algorithm due to abrupt movements. The abrupt amplitude in the initial iterations and the slight amplitude in the subsequent iterations will ensure the rapid convergence of NLBGND and an efficient search close to the global optimum [20] and also encourage NLBGND's change from exploratory to exploitative patterns. In addition, this analysis will display the behavior of the proposed NLBGND for exploration. Reviewing the average fitness curve presented in Figs. 4–6, improvements in the NLBGND fitness during iterations have shown the positive influence of the Levy flight, Brownian motion and nonlinear control variable. While the average fitness of NLBGND is detracted in some situations, the curve shows a steady descent, which is a sign of enhancing the overall output of the population.

D. Scalability Analysis

In this section, the scalability of our proposed NLBGND algorithm was reviewed and analyzed accordingly. the scalability assessment is used to analyze the influence of different dimensions on the efficiency of NLBGND. It also reveals how a meta-heuristic can maintain its higher-dimensional search strengths. The stochastic nature of the meta-heuristics allows the effects of a single run to be inaccurate so that all algorithms are tested 30 times with different set of parameters. Also, the statistical measurements are collected; mean, in addition to capturing the standard deviations and all related results shown in Tables 6–9. For each function, the observed mean (Ave) and standard deviation (Std) results of all optimization algorithms for 30 runs of 500 iterations were gathered and evaluated.

1) Exploitation capability evaluation

The results presented in Table 6 clearly indicate that NLBGND placed first or tied first on average when solving F1–F7. Results demonstrate that NLBGND has been able to surpass other methods, particularly GNDO, in almost all test functions. This capability is obtained from the small movements defined by Lévy flight function in the proposed exploitation phase, and a nonlinear parameter that adjusts the phases of exploitation and exploration of NLBGND.

TABLE 6. RESULTS OF BENCHMARK FUNCTIONS (F1–F7), WITH 30 DIMENSIONS.

Func.	MV O	MFO	SSA	GW O	PSO	DE	WO A	GN DO	LG ND
F1	A 1.27	1.03	4.86	2.97	2.60	1.38	2.72	4.09	1e-
	ve E+01	E+04	E+00	E-18	E-01	E+00	E-76	E+02	323
	St 3.44	1.02	4.25	4.67	3.18	4.67	7.18	4.27	0.00
	d E+00	E+04	E+00	E-18	E-01	E-01	E-76	E+02	E+00
F2	A 2.14	8.19	1.26	1.56	2.27	2.75	9.23	6.29	1.24e
	ve E+05	E+01	E+01	E-11	E+00	E-01	E-50	E+00	-322
	St 1.17	3.60	3.60	8.26	9.96	4.58	2.68	1.97	0.00
	d E+06	E+01	E+00	E-12	E-01	E-02	E-49	E+00	E+00
F3	A 6.42	6.21	1.15	3.97	1.80	1.09	4.40	1.52	3.00e
	ve E+03	E+04	E+04	E-01	E+03	E+05	E+04	E+03	-323
	St 1.57	2.17	5.33	7.72	3.69	1.14	1.09	7.05	0.00
	d E+03	E+04	E+03	E-01	E+02	E+04	E+04	E+02	E+00
F4	A 2.15	8.60	2.16	9.27	4.48	4.62	5.65	1.85	2.00e
	ve E+01	E+01	E+01	E-04	E+00	E+01	E+01	E+01	-323
	St 8.37	3.50	3.58	8.02	9.09	3.95	2.58	3.15	0.00
	d E+00	E+00	E+00	E-04	E-01	E+00	E+01	E+00	E+00
F5	A 9.24	1.93	2.78	4.77	4.52	1.27	2.80	6.35	5.77
	ve E+02	E+07	E+03	E+01	E+02	E+03	E+01	E+04	E+00
	St 7.98	3.98	5.13	6.42	2.51	4.51	4.62	9.15	4.38
	d E+02	E+07	E+03	E-01	E+02	E+02	E-01	E+04	E-01
F6	A 1.19	1.11	4.67	3.10	3.49	1.32	3.09	3.57	6.27
	ve E+01	E+04	E+00	E+00	E-01	E+00	E-01	E+02	E+00
	St 3.35	1.04	4.66	6.36	4.09	4.23	1.83	2.12	4.62
	d E+00	E+04	E+00	E-01	E-01	E-01	E-01	E+02	E-01
F7	A 1.32	1.83	6.48	4.18	3.43	1.95	3.74	3.64	4.14
	ve E-01	E+01	E-01	E-03	E+00	E-01	E-03	E-01	E-04
	St 4.00	2.02	2.10	2.10	2.74	3.37	4.77	1.92	4.23
	d E-02	E+01	E-01	E-03	E+00	E-02	E-03	E-01	E-04

2) Exploration capability evaluation

The results reported in Table 7 shows that NLBGND is still efficient in multimodal functions. For F10–F13, the AVG of the proposed method has been the smallest compared towards the other algorithms. Moreover, the results of NLBGND in F8–F9 are desirable and just slightly smaller than WOA, suggesting that NLBGND can still retain its superiority over advanced algorithms and illustrate the capacity of NLBGND to avoid local optimal solution. According to the results obtained, it can

be concluded that the addition of Brownian motion theorem to the GNDO algorithm not only enhances the exploration phase of this algorithm, but also strengthens the efficiency of the algorithm.

TABLE 7. RESULTS OF BENCHMARK FUNCTIONS (F8–F13), WITH 30 DIMENSIONS.

	Func.	MVO	MFO	SSA	GWO	PSO	DE	WOA	GNDO	LGND o
F 8	A V G	- 1.22 E+04	- 1.32 E+04	- 1.21 E+04	- 9.28 E+03	- 5.88 E+03	- 1.21 E+04	- 1.07 E+04	- 5.92 E+03	- 2.27 E+03
	S T D	9.55 E+02	1.47 E+03	8.98 E+02	1.03 E+03	1.97 E+03	5.57 E+02	1.75 E+03	1.29 E+03	6.40 E+02
	A V G	2.74 E+02	3.59 E+02	1.01 E+02	6.01 E+00	1.63 E+02	2.47 E+02	0.00 E+00	3.07 E+01	5.68 E-14
F 9	S T D	4.70 E+01	6.06 E+01	2.63 E+01	7.23 E+00	2.68 E+01	1.08 E+01	0.00 E+00	9.46 E+00	0.00 E+00
	A V G	4.08 E+00	1.96 E+01	5.50 E+00	2.08 E-10	1.71 E+00	4.54 E-01	4.80 E-15	9.37 E+00	4.44 E-15
F 10	S T D	4.28 E+00	4.52 E-01	1.39 E+00	8.80 E-11	4.68 E-01	1.42 E-01	2.85 E-15	1.15 E+00	0.00 E+00
	A V G	1.11 E+00	1.12 E+02	9.35 E-01	3.17 E-03	2.35 E-02	8.32 E-01	3.70 E-12	5.16 E+00	1.11 E-16
F 11	S T D	2.51 E-02	9.33 E+01	1.89 E-01	7.38 E-03	1.33 E-02	9.26 E-02	2.03 E-10	2.44 E+00	0.00 E+00
	A V G	6.47 E+00	1.14 E+07	1.41 E+01	1.33 E-01	1.00 E-01	1.45 E+00	9.93 E-01	7.92 E+00	3.05 E-02
F 12	S T D	2.56 E+00	4.67 E+07	5.43 E+00	4.32 E-02	1.28 E-01	5.90 E-01	2.35 E-01	7.62 E+00	2.31 E-02
	A V G	9.69 E+00	4.91 E+07	8.50 E+01	3.35 E+00	4.31 E-01	3.06 E+00	4.79 E-01	4.04 E+03	9.32 E-02
F 13	S T D	1.01 E+01	1.25 E+08	1.48 E+01	2.78 E-01	3.39 E-01	1.22 E+00	2.38 E-01	7.75 E+03	3.44 E-02

In contrast, the results in Table 8 indicate that NLBGND0 had very promising result in coping with F14-F23 events. However, with other methods, the best global results for F14-F23 can still be obtained. From the results derived, it can be concluded that if there is a high exploration and exploitation rate and a reasonable balance between the two phases of exploration and exploitation in meta-heuristic algorithms, those algorithms can do well in the dealing with optimization problems. The NLBGND0 algorithm profits from the strengths of Levy Flight and Brownian motion to improve the phases of exploitation and exploration. Moreover, by using a nonlinear control parameter, it attempts to create an equilibrium between the two phases of exploration and exploitation. Based on the obtained results by the proposed method relative to other methods, the strategy used in the NLBGND0 algorithm has been demonstrated to be advantageous.

It can be very difficult for metaheuristic optimization algorithms to cope with composite mathematical functions because of their intrinsic challenge. Solving those issues requires an appropriate balance between the processes of exploration and exploitation. As reported in Table 9, NLBGND0 appears to be stronger than other algorithms. This shows the effectiveness of using the control parameter in the proposed algorithm has tended to a good balance between exploration and exploitation phases.

On the other hand, the run-time of meta-heuristic algorithms to find an optimum solution is a critical consideration that needs to be addressed in order to assess our proposed algorithm against such a metric. To examine the efficiency of the

proposed method and to verify the impact of the use of Levy flight and Brownian motion on its running time, all optimizing running times for F1–F13 optimization problems with 1000 dimensions are reported and detailed in table 10. The Wilcoxon rank-sum test was carried out to statistically analyze running time. Wilcoxon rank-sum is a non-parametric value used in the performance evaluation method as a complement to the two-sample t-test, which is based purely on the order in which the results from the two samples drop. This procedure can be used to determine the effects of the proposed method in pairs with other techniques.

TABLE 8. RESULTS OF BENCHMARK FUNCTIONS (F14–F23).

Func		GWO	PSO	DE	MVO	SSA	MFO	WOA	GND O	LGN DO
F 1 4	A ve	3.67 E+00	1.23 E+00	1.02 E+00	9.98 E-01	1.16 E+00	1.86 E+00	2.93 E+00	1.06 E+00	3.25 E+00
	St d	4.05 E+00	4.28 E-01	1.28 E-01	1.71 E-11	4.58 E-01	1.48 E+00	2.97 E+00	2.52 E-01	3.57 E+00
	A ve	2.37 E-03	7.46 E-04	1.34 E-03	4.02 E-03	1.59 E-03	2.35 E-03	1.06 E-03	1.64 E-03	2.28 E-03
F 1 5	St d	6.10 E-03	2.51 E-04	4.10 E-03	7.44 E-03	3.56 E-03	4.91 E-03	1.34 E-03	5.09 E-03	8.43 E-03
	A ve	- E+00	- E+00	- E+00	- E+00	- E+00	- E+00	- E+00	- E+00	- E+00
	St d	4.60 E-09	6.78 E-16	2.47 E-02	2.87 E-07	2.21 E-14	6.78 E-16	9.35 E-10	6.83 E-15	4.72 E-04
F 1 7	A ve	3.98 E-01	3.98 E-01	3.99 E-01	3.98 E-01	3.98 E-01	3.98 E-01	3.98 E-01	3.98 E-01	5.42 E-01
	St d	1.02 E-04	0.00 E+00	6.16 E-03	2.91 E-07	5.65 E-15	0.00 E+00	1.06 E-05	5.06 E-12	9.44 E-02
	A ve	3.00 E+00	3.00 E+00	3.07 E+00	3.00 E+00	3.00 E+00	3.00 E+00	3.00 E+00	3.00 E+00	3.02 E+00
F 1 8	St d	3.19 E-06	4.81 E-16	5.98 E-01	1.76 E-06	2.85 E-13	1.53 E-15	1.47 E-04	2.43 E-15	1.19 E-01
	A ve	3.86 E+00	3.86 E+00	3.86 E+00	3.86 E+00	3.86 E+00	3.86 E+00	3.85 E+00	3.86 E+00	3.81 E+00
	St d	2.40 E-03	2.68 E-15	8.55 E-03	1.06 E-06	5.18 E-09	1.44 E-03	1.80 E-02	2.22 E-15	3.93 E-02
F 2 0	A ve	3.27 E+00	3.28 E+00	3.30 E+00	3.25 E+00	3.22 E+00	3.23 E+00	3.20 E+00	3.28 E+00	2.98 E+00
	St d	6.00 E-02	5.83 E-02	7.24 E-02	5.96 E-02	5.54 E-02	5.65 E-02	1.78 E-01	5.70 E-02	9.69 E-02
	A ve	9.48 E+00	7.88 E+00	9.46 E+00	8.47 E+00	7.65 E+00	6.56 E+00	8.26 E+00	8.16 E+00	5.31 E+00
F 2 1	St d	1.75 E+00	2.68 E+00	1.55 E+00	2.67 E+00	3.41 E+00	3.33 E+00	2.50 E+00	3.36 E+00	5.55 E-01
	A ve	1.04 E+01	9.52 E+00	9.92 E+00	8.66 E+00	8.20 E+00	6.60 E+00	7.60 E+00	7.44 E+00	4.69 E+00
	St d	4.40 E-04	2.00 E+00	1.61 E+00	2.77 E+00	3.23 E+00	3.65 E+00	3.12 E+00	3.70 E+00	1.17 E+00
F 2 3	A ve	9.99 E+00	9.82 E+00	1.00 E+01	8.86 E+00	9.10 E+00	7.87 E+00	7.55 E+00	8.10 E+00	4.96 E+00
	St d	2.06 E+00	1.85 E+00	1.53 E+00	3.13 E+00	2.96 E+00	3.39 E+00	3.30 E+00	3.53 E+00	3.49 E+00

TABLE 9. RESULTS FOR COMPOSITION FUNCTIONS (CEC2014) [9].

Benchmark		GW O	PSO	DE	MV O	SSA	MFO	WO A	GND O	LGN DO
F24	Ave	2.64 E+03	2.61 E+03	2.62 E+03	2.62 E+03	2.63 E+03	2.67 E+03	2.69 E+03	2.68 E+03	2.50 E+03
	Std	1.10 E+01	8.66 E-05	1.72 E-03	5.46 E+00	9.15 E+00	3.93 E+01	2.23 E+01	9.59 E+01	0.00 E+00
	Ave	2.60 E+03	2.62 E+03	2.63 E+03	2.64 E+03	2.64 E+03	2.68 E+03	2.61 E+03	2.65 E+03	2.60 E+03
	Std	1.12 E-02	7.36 E+00	1.70 E+00	7.08 E+00	6.59 E+00	2.98 E+01	8.25 E+00	1.08 E+01	0.00 E+00
F26	Ave	2.71 E+03	2.72 E+03	2.72 E+03	2.71 E+03	2.72 E+03	2.72 E+03	2.72 E+03	2.72 E+03	2.70 E+03
	Std	6.00 E+00	4.12 E+00	3.00 E+00	2.26 E+00	4.86 E+01	1.01 E+01	2.03 E+01	3.66 E+00	0.00 E+00
	Ave	2.74 E+03	2.77 E+03	2.70 E+03	2.74 E+03	2.70 E+03	2.71 E+03	2.73 E+03	2.75 E+03	2.72 E+03
	Std	4.96 E+01	4.48 E+01	7.86 E-02	5.92 E+01	1.38 E-01	4.95 E+01	6.52 E+01	5.05 E+01	2.78 E+01
F27	Ave	3.43 E+03	3.59 E+03	3.49 E+03	3.35 E+03	3.60 E+03	3.63 E+03	3.85 E+03	3.76 E+03	2.90 E+03

F	St	1.32	2.72	1.27	1.68	1.73	1.84	3.81	2.10	8.44
2	d	E+02	E+02	E+02	E+02	E+02	E+02	E+02	E+02	E-14
F	A	4.04	6.72	3.73	4.10	4.14	3.92	5.43	4.44	3.00
2	ve	E+03	E+03	E+03	E+03	E+03	E+03	E+03	E+03	E+03
9	St	3.24	6.03	3.65	3.45	3.60	1.22	8.53	3.04	8.44
	d	E+02	E+02	E+01	E+02	E+02	E+02	E+02	E+02	E-14

TABLE 10. COMPARISON OF AVERAGE RUNNING TIME RESULTS (SECONDS) OVER 30 RUNS.

Benchmark		MV O	MFO	SSA	GW O	PSO	DE	WO A	GND O	LGN DO
F1	Ave	1.70 E+01	8.14 E+00	7.15 E+00	1.42 E+01	5.40 E+00	4.81 E+00	5.79 E+00	7.90 E-01	1.09 E+00
	Std	1.05 E+00	3.80 E-01	4.03 E-01	6.18 E-01	4.78 E-01	3.52 E-01	2.35 E-01	1.19 E-01	8.70 E-02
	Dev	8.00 E+00	7.59 E+00	6.42 E+00	1.22 E+01	4.62 E+00	4.34 E+00	7.32 E+00	7.44 E-01	1.04 E+00
F2	Ave	4.80 E-01	4.13 E-01	2.50 E-01	2.04 E-01	2.32 E-01	2.34 E-01	5.75 E-01	1.08 E-01	9.32 E-02
	Std	7.75 E+01	6.91 E+01	6.91 E+01	7.69 E+01	6.75 E+01	6.78 E+01	7.30 E+01	2.04 E+00	2.27 E+00
	Dev	3.57 E+00	3.30 E+00	3.76 E+00	3.60 E+00	3.85 E+00	3.87 E+00	2.66 E+00	3.73 E-01	1.64 E-01
F3	Ave	1.59 E+01	7.73 E+00	1.16 E+02	1.30 E+01	5.06 E+00	4.43 E+00	6.29 E+00	7.39 E-01	1.02 E+00
	Std	1.26 E+00	5.22 E-01	2.08 E+00	1.15 E+00	4.21 E-01	4.74 E-01	5.71 E-01	1.15 E-01	9.02 E-02
	Dev	1.76 E+01	1.54 E+01	6.67 E+00	2.03 E+01	8.98 E+00	4.29 E+00	8.84 E+00	8.97 E-01	1.19 E+00
F5	Ave	2.94 E-01	2.22 E+00	1.51 E-01	3.32 E+00	2.14 E-01	1.23 E-01	2.87 E-01	1.58 E-01	1.41 E-01
	Std	1.55 E+01	1.25 E+01	6.46 E+00	1.77 E+01	6.74 E+00	4.27 E+00	5.90 E+00	8.02 E-01	1.13 E+00
	Dev	3.83 E-01	9.04 E-01	1.11 E-01	1.19 E+00	3.09 E-01	1.38 E-01	2.30 E-01	1.52 E-01	1.25 E-01
F7	Ave	2.50 E+01	2.35 E+01	1.14 E+01	2.83 E+01	1.66 E+00	9.76 E+01	1.61 E+01	1.33 E+00	1.66 E+00
	Std	2.94 E-01	1.31 E+00	3.32 E-01	1.41 E+00	2.35 E-01	2.02 E-01	4.54 E-01	1.00 E-01	1.53 E-01
	Dev	1.51 E+01	1.59 E+01	8.28 E+00	2.14 E+01	1.06 E+01	6.34 E+00	8.91 E+00	6.94 E-01	1.46 E+00
F8	Ave	2.61 E-01	1.18 E+00	2.37 E-01	1.47 E+00	5.42 E-01	1.07 E-01	5.09 E-01	6.09 E-02	5.81 E-02
	Std	1.73 E+01	1.35 E+01	7.50 E+00	1.71 E+01	8.78 E+00	6.17 E+00	6.91 E+00	7.94 E-01	1.06 E+00
	Dev	2.95 E-01	3.72 E-01	2.45 E-01	3.06 E-01	6.89 E-01	1.35 E-01	6.50 E-01	8.09 E-02	7.81 E-02
F9	Ave	1.77 E+01	1.41 E+01	7.58 E+00	1.75 E+01	8.84 E+00	6.30 E+00	7.29 E+00	8.32 E-01	1.06 E+00
	Std	2.99 E-01	2.65 E-01	2.57 E-01	4.17 E-01	5.77 E-01	1.27 E-01	5.66 E-01	7.26 E-02	8.08 E-02
	Dev	2.03 E+01	1.73 E+01	8.78 E+00	2.12 E+01	1.06 E+01	6.86 E+00	9.28 E+00	9.30 E-01	1.16 E+00
F11	Ave	1.03 E+00	6.17 E-01	2.96 E-01	5.05 E-01	7.18 E-01	1.27 E-01	5.78 E-01	1.00 E-01	6.45 E-02
	Std	4.28 E-01	4.18 E-01	1.82 E-01	4.54 E-01	3.90 E+01	1.66 E+01	3.79 E+01	2.43 E+00	2.72 E+00
	Dev	2.95 E-01	5.98 E-01	2.85 E-01	4.04 E-01	2.25 E+00	3.23 E-01	3.02 E+00	1.90 E-01	1.34 E-01
F12	Ave	4.45 E+01	4.18 E+01	1.81 E+01	4.57 E+01	3.95 E+01	1.67 E+01	3.80 E+01	2.44 E+00	2.65 E+00
	Std	1.88 E+00	3.62 E-01	2.70 E-01	4.33 E-01	1.77 E+00	2.73 E-01	1.45 E+00	1.12 E-01	8.13 E-02

Reviewing Table 10, it is noticeable that the NLBGND0 has an adequate running time. However, the application of Levy's flight and Brownian motions had no impact on the time needed to find the best global approach compared to GND0. Observing the collected results, the NLBGND0 algorithm could obtain fair and competitive running time relative to other optimizers in the handling of unimodal and multimodal functions and high-dimensional solution search space. This success is due to the special structure of meta-heuristics in handling with the optimization problems.

Due to the p-values reported in Table 11, it is apparent that NLBGND0 demonstrates considerably superior performance than other optimizers in all the cases assessed. The results achieved by our proposed algorithm are greatly better relative to those recorded by other approaches. In neither case, the other

strategies were not capable of providing the results achieved by the proposed NLBGND0.

TABLE 11. RESULTS OF WILCOXON RANK-SUM TEST OVER ALL RUNS.

Fun	WOA	MVO	SSA	MFO	GWO	PSO	DE	GND
c.								O
F1	2.62E-11	2.62E-11	2.62E-11	2.62E-11	2.62E-11	3.41E-20	2.62E-11	2.62E-11
F2	2.39E-11	2.39E-11	2.39E-11	2.39E-11	2.39E-11	3.41E-20	2.39E-11	2.39E-11
F3	1.18E-12	2.96E-11	2.96E-11	2.96E-11	1.18E-12	3.41E-20	2.96E-11	2.96E-11
F4	2.51E-11	2.51E-11	2.51E-11	2.51E-11	2.51E-11	3.41E-20	2.51E-11	2.51E-11
F5	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	3.41E-20	3.02E-11	3.02E-11
F6	3.02E-11	3.02E-11	3.02E-11	5.20E-01	3.02E-11	3.41E-20	3.02E-11	3.02E-11
F7	3.52E-07	5.60E-07	3.02E-11	3.02E-11	1.86E-09	1.89E-21	3.02E-11	3.02E-11
F8	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	2.65E-06	3.02E-11	4.08E-11
F9	1.69E-14	1.21E-12	1.21E-12	1.21E-12	1.19E-12	3.38E-12	1.21E-12	1.21E-12
F10	1.69E-14	1.21E-12	1.21E-12	6.14E-14	1.21E-12	3.39E-20	1.21E-12	1.21E-12
F11	5.86E-11	1.21E-12	1.21E-12	1.21E-12	9.70E-06	3.39E-20	1.21E-12	1.21E-12
F12	1.11E-06	3.02E-11	9.18E-06	7.94E-03	3.02E-11	3.41E-20	3.02E-11	4.50E-11
F13	3.02E-11	3.02E-11	3.02E-11	8.48E-09	3.02E-11	3.41E-20	3.02E-11	3.02E-11

On the other hand, these results demonstrate that the use of levy flight and Brownian motion values and a nonlinear control parameter seems to have a positive impact on the efficiency of the proposed method. In addition, the results of Wilcoxon's rank-sum tests, summarized in Table 11, have shown that NLBGND0 has the potential to outperform other strategies with a considerable amount of difference. It is worth noting that NLBGND0 has not only attained a lower global optimum in nearly all cases than other strategies, but also has a high output relative to GND0 in handling these fitness functions.

In order to assess the convergence performance of our proposed NLBGND0 algorithm, the best score obtained so far has been defined and estimated as the optimal value achieved. This score was measured and reported for each individual iteration of all the test functions used. This value was determined using each of the benchmarked methods and checked thoroughly along with the average results of our proposed NLBGND0, which recorded from the run of 30 simulations with 500 iterations for each. In the discussion on convergence, as seen in Fig. 7, in most situations, the proposed method has a reasonable convergence rate relative to other approaches. The observed results have indicated and asserted the reliability of the use of levy flight and Brownian motion, which has significantly improved the efficiency and the convergence rate of the optimization algorithm directly presented by our proposed process. However, with a few of the functions, NLBGND0 obtained the optimum values that were nearly achieved by the GND0 norm. The NLBGND0 algorithm not only surpassed the other algorithms in the convergence curve, but also reached an optimum value in initialization step and demonstrated a very impressive convergence speed relative to other algorithms.

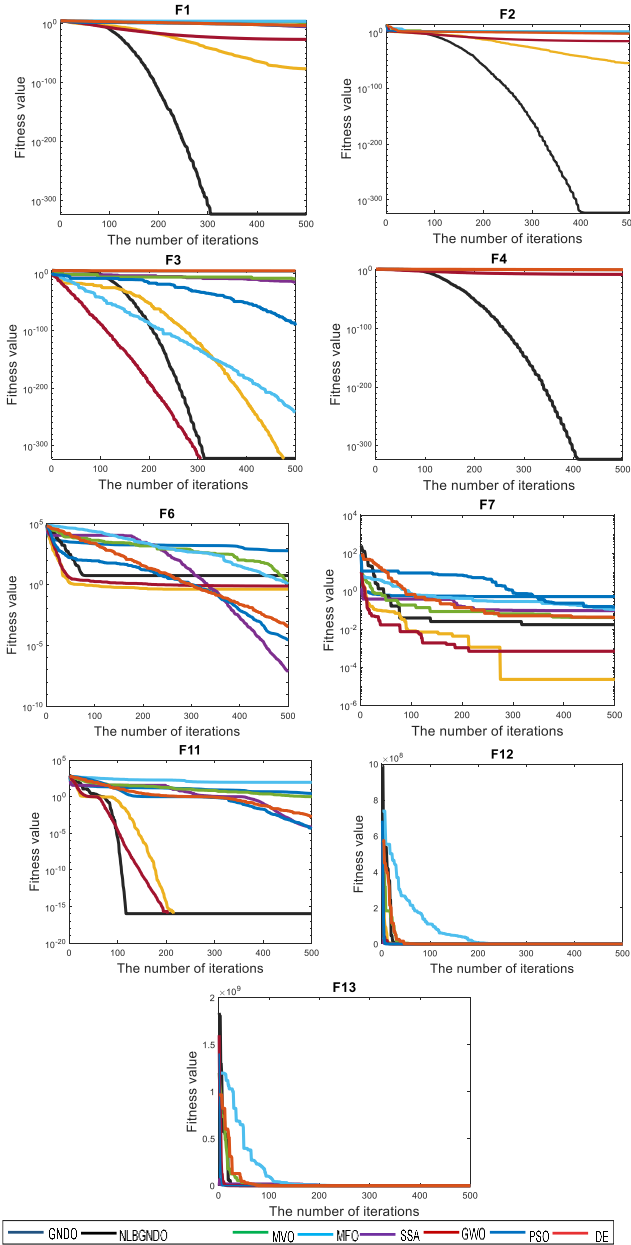


Fig.7. Convergence analysis of the NLBGND algorithm compared with other techniques.

E. Sensitivity Analysis

NLBGND, like every other new algorithm, has several control parameters. Analysis of certain keys, internal parameters, such as the number of search agents, the maximum number of iterations, and the control parameter defined in Eq. (19) is necessary in order to achieve optimal efficiency. A sensitivity analysis is performed in this sub-section to examine the effect of the key control parameters of the proposed method on its efficiency. The following values shall be used:

- Number of search agents (N): 30, 50, 80, 100.
- Maximum iteration number (T): 50, 100, 500, or 1000.

- Nonlinear control parameter c: Nonlinearly decreases from 1 to 0, 2 to 0, 3 to 0, or 4 to 0.

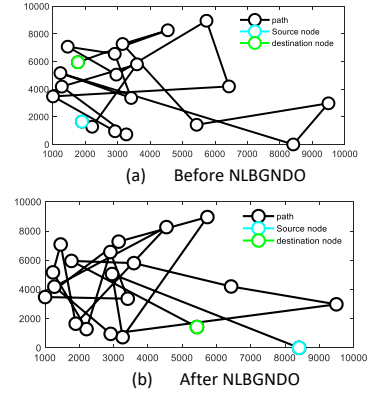


Fig. 8. Path selection among 20 mobile sensor nodes with Graph size 10kx10k.

In order to indicate the influence of these parameters on the efficiency of NLBGND, three different experiments were performed concurrently by adjusting the values of the three parameters mentioned earlier. To be able to envision the efficiency surface of each experiment, one of the parameters is assumed constant while the other two are changed. NLBGND is evaluated 30 times on four test functions (F1, F7, F12, and F13) and the last iteration average of the best solution is used to simulate the average performance of the surface in Fig. 7. Fig. 9 shows the results of parameter setting analysis.

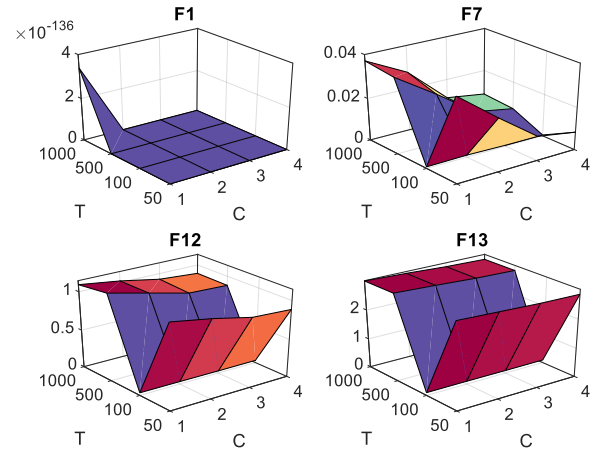


Fig. 9. Parameter setting surface indicating the best combination of C and T controlling parameters.

V. EFFICIENT AND TRUSTWORTHY ROUTING ALGORITHM FOR IOT-FINTECH USING NLBGND

In this section the problem formulation of routing algorithm for a supply-chain application for IoT-FinTech will be presented and discussed in detail. This will be followed by the implementation, results, and discussion.

A. Problem Formulation

In mobile Ad Hoc sensor networks, identifying the optimal path from source to a destination with such highly distributed and dynamic topology is a challenging task. Adding to this, security and trust are considered other key challenges while

formulating a decision on path selection to be used in routing data over IoT network, especially when this type of data is considered as sensitive as it belongs to FinTech applications. On the other hand, as discussed earlier, having an optimal solution that considering all these aspects is a daunting task. Therefore, in this section we have presented a routing protocol based on the proposed NLBGND optimization algorithm to be used in adaptively formulate paths to be used for data forwarding over IoT network.

We first started by formulating the objective function to be used by our NLBGND algorithm and the other benchmark methods. Equation 19 demonstrates the main function that will be used in each time a node trying to formulate a routing path between source i to a destination j .

$$f(O_{ijn}) = \frac{[T_{Response(ij)n}]^{\alpha} [R_{Energy(ij)n}]^{\sigma} [F_{Trust(ij)n}]^{\gamma} [D_{Inter.n}]^{\rho}}{\sum_{P \in G(i,j)} [T_{Response(ij)n}]^{\alpha} [R_{Energy(ij)n}]^{\sigma} [F_{Trust(ij)n}]^{\gamma} [D_{Inter.n}]^{\rho}} \quad (20)$$

Where $T_{Response_{ijn}}$ is the response time experienced by node i in sending hello message for neighboring nodes under its coverage area and getting back the response. This time will be calculated instantly for each node and hold temporally in a vector to be used during the selection process. While $R_{Energy_{ijn}}$ is the advertised residual energy by each node within the coverage range of node i to be used in transmitting data to node j and $F_{Trust_{ijn}}$ is the trust value of each node that will be calculated based on a proposed friendship mechanism. Additionally, to consider the sensor nodes density in the surrounding area of a node in the selection process, inter node distance factor will be calculated $D_{Inter.n}$. These metrics will be gathered for each individual node that involved in the IoT's supply-chain network then they will be multiplied together to produce one factor to be divided by the total attributed value measured for each entire path P belongs to the graphed network. Fig. 8 demonstrating a simple example that consisting of 20 mobile sensor nodes in the developed simulation using Matlab, where source and destination nodes are highlighted accordingly. It is important mentioning that Fig. 8 (a) showing the initial selected path between source and destination, while Fig. 8 (b) showing the selected optimal path between them according to Equation 19 using our proposed NLBGND optimization algorithm.

In order to consider consistency in decision-making and avoiding the negative impact of such metric that produces unreliable behavior due to unstable environmental/technical characteristics of a sensor node, we have introduced the use of α, σ, γ and ρ coefficients. Where $0 \geq \alpha, \sigma$ and $\gamma \leq 1$, these values will be calculated and defined according to the statistical analysis of each of the above defined metric. For instance, when the standard deviation (sd) of $T_{Response_{ijn}}$ shows that 50% of its data within the *min* and *max* of the index value of sd ($sd = \frac{max-min}{4}$), this indicating that $T_{Response_{ijn}}$ its half way reliable enough to orient the value of O_{ijn} according to it. Hence the coefficient value of $\alpha = 0.5$, the same process will be applied for the other coefficients, except γ that is belongs to the friendship trust metric as when it has bad sd value that reflects untreated nodes or path; so, the value of zero should be

considered to exclude this node/path. Equation 20 present the response time calculation for the entire path P along with the constrains. According to [32], the maximum tolerable end-to-end delay for most of IoT related real-time application should be not more than 100ms. Therefore, we have set the constrains accordingly to return a value zero (which means best value of $T_{Response_{ijn}}$) when the obtained response time for path p is within a range not more than 100ms, value 0.5 when it goes beyond 100ms but not more than 500ms, which is tolerable for other non-real-time applications and value 1 otherwise.

$$T_{Response(ij)n} = \sum_{P \in G(i,j)} T_{Response(ij)n} \quad (21)$$

$$T_{Response_{ijn}} = \begin{cases} 0 & \text{if } 0 \geq T_{Response_{ijn}} \leq 100ms \\ 0.5 & \text{if } 100 \geq T_{Response_{ijn}} \leq 500ms \\ 1 & \text{otherwise} \end{cases}$$

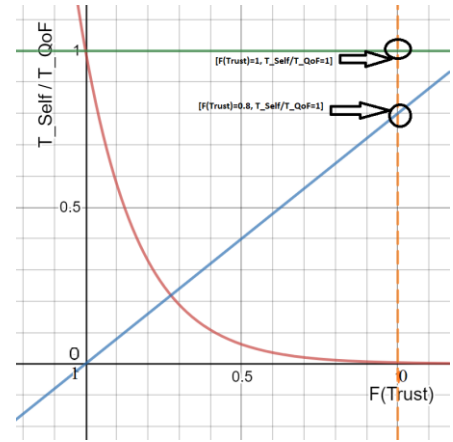


Fig. 10. Impact of T_{Self} and T_{QoF} on friendship trust value for routing protocol in IoT-FinTech.

On the other hand, Equation 21 shows the calculation of residual energy of each available path P within a graph of mobile sensors within the IoT network. The range of this metric will be 0-100 joule. The residual energy will be obtained for each node and inserted in a set, then the minimum value obtained by n node will be the constrain of selecting such path that involving such node with limited energy, as this might lead eventually for the entire service to be disturbed or terminated.

$$R_{Energy_{ijn}} = \text{Min} \{R_{Energy_{ijn}}\} \quad (22)$$

$$R_{Energy_{ijn}} = \begin{cases} 1 & \text{if } 80 \geq R_{Energy_{ijn}} \leq 100 \text{ joule} \\ 0.5 & \text{if } 20 \geq T_{Response_{ijn}} \leq 80 \text{ joule} \\ 0 & \text{otherwise} \end{cases}$$

To address the trust and security aspects, we have formulated Equation 22 based on the inspiration from a study presented in [33], which presented the effects of interpersonal influence regulation on the quality of relationships. Hence, this paper we have considered two factors of trust in forming a reliable

friendship between sensor nodes for IoT-FinTech applications. First, it is the self-evaluation T_{self} of trust based on experience with a particular node n . While the second factor is the trust based on Quality of Friendship T_{QoF} of a sensor node by other experienced nodes within the IoT network that had interacted previously with.

$$F_{Trust_{ijn}} = \min\{T_{self}, T_{QoF}\} \quad \forall \text{Nodes} \in P_{ij} \quad (23)$$

$$F_{Trust_{ijn}} = \begin{cases} 1 & \text{if } 0.8 \geq T_{self} \text{ OR } T_{QoF} \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

The constrains here showing that this factor will only obtain a full score when the minimum value of any of the nodes within a path P is not less than 0.8, otherwise it will be given zero. This will enable only high trustworthy nodes to be participating in the routing path formulation for transmitting data related to FinTech within the IoT network. Fig. 10 demonstrates the impact of having either T_{self} or $T_{QoF} = 0.8$ or 1 that obtained from any sensor within a route option, the F_{Trust} value will set to 1, otherwise will be slimmed to zero. We can also observe from Fig. 10 that the nature of F_{Trust} will exponentially decreased to the decreasing with T_{self} or T_{QoF} , it will be 1 when the obtained level of trust within a node or other neighbouring nodes to the one in the selection process is high. In another words, the smallest the variation with T_{self} and the higher is the T_{QoF} the more F_{Trust} will be likely in an acceptable level.

The last factor that was considered in our problem formulation was the nodes inter-nodes distance D_{Inter_n} . We have adopted the concept of formulating this metric from our previously published work that related to Internet-of-Vehicles [34]. Readers may refer to section 3 in that study to understand the insight of formulating D_{Inter_n} .

To aggregate all these four metrics in one singly minimization objective function, we have multiplied $R_{Energy_{ijn}}$ and $F_{Trust_{ijn}}$ with value of (-1) to ensure that all metrics that contributing to the objective function are working on same direction, (the lowest the value is the best).

B. Results and Analysis

To prove the concept of the competency of our proposed NLBGND0 optimization algorithm in solving complex real-life problems, the formulated optimization problem in the previous sub-section has been implemented and tested with NLBGND0 algorithm. Fig. 11 demonstrating the convergence rates in finding the optimal path from source to destination nodes using our proposed NLBGND0 algorithm alongside with other well-known optimization algorithms.

It has been also witnessed that our proposed NLBGND0 algorithm could outperform others in finding the optimal path/route with minimum cost and risk to route data traffic related to IoT-FinTech applications. The other interesting point to mention here, that the convergence rate of our proposed NLBGND0 algorithm was in a pretty shape that shows the algorithm was converging in wise steps towards the optimal solution/path from source destination nodes.

This reflecting the fact that the introduced modification to the native GND0 algorithm (adding non-linear and Lévy Brownian

parameters) has contributed significantly to obtaining the best tune between exploration and exploitation phases.

Another observation was that when the landscape of the IoT graph has scaled to 10000 and the number of deployed sensors was 20, ACO algorithm had better convergence after about 80 iteration from the starting point in comparison with other algorithms (including our NLBGND0). The fact is that when in graph size getting bigger, the sparsity of sensor nodes will be much higher, which will make it challenging task for most of searching/optimization algorithms. To comment here, ACO by its nature working great in exporting problem's space, due to its nature in following the pheromone communication concept of biological ants. Through, relying only on this factor did not lead to the optimal route to be used in forwarding IoT-FinTech's data. As we can see from Fig 11 (f), ACO could not eventually find the optimal value as it has missed some of the promising sensor nodes that could give better route formulation in forwarding data between source and destination nodes. While in contrast, our proposed NLBGND0 could managed eventually to find the best path that could produce zero cost and risk.

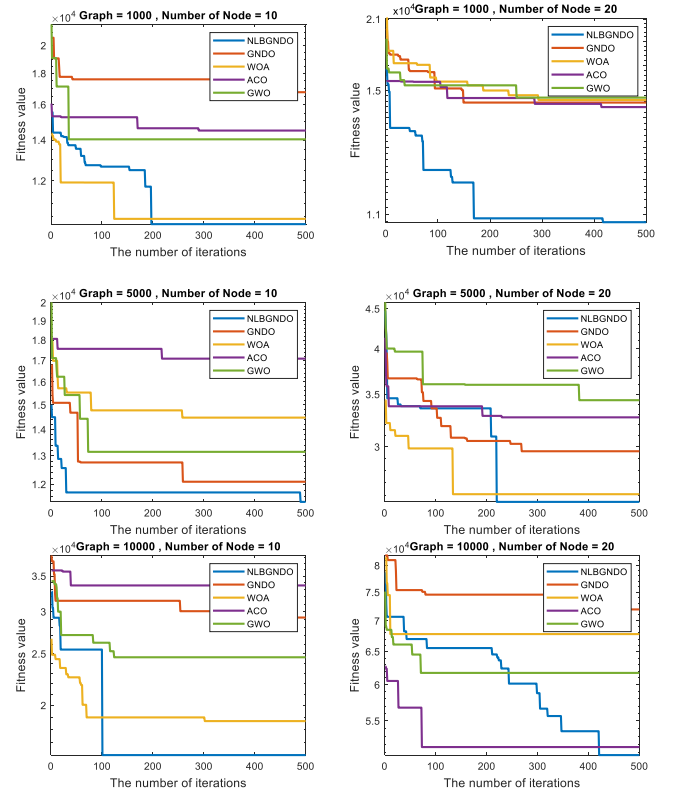


Fig. 11. Convergence curves of minimizing cost and risk for routing in IoT-FinTech.

We could also observe from the behaviour of our proposed algorithm with 20 mobile sensor nodes that its ability in finding the optimal path to destination has been increased. As discussed earlier, that was due to the increasing with the number of the possible solutions that contributes to forming more alternative paths due to nodes' density. Taking the benefits of the improvised version of GND0 with Levy and Brownian strategies along with the non-linear controlling parameter, (as

discussed earlier in Section III), our NLBGND algorithm will provide more promising results with high dense scenarios. This gives our solution more potential for future scalability with relatively high number of mobile sensors to be deployed within the IoT network.

VI. CONCLUSION

In this paper a new trustworthy and efficient mechanism has been proposed, which helps in finding the optimal path to be used in forwarding the application's generated data traffic over IoT for FinTech applications. The paper has first investigated the state of the arts' limitations in considering multi-objective optimization problem for routing techniques in IoT networks. We have found that considering smart enough optimization algorithm in finding the optimal solution in efficient and effective way is the key to solve such challenging problem. Therefore, we have developed a Non-linear Lévy Brownian Generalized Normal Distribution Optimization (NLBGND) optimization algorithm that levitates the performance and quality of the selected solutions of a given problem. This new proposed algorithm has been first tested with a wide range of benchmark functions to measure and validate its ability in finding the global optimal solution among the available set of local optimal solutions. Afterwards, a routing problem for IoT-FinTech applications was formulated considering four essential metrics that aiming to shape an optimal trustworthy and efficient route between source and destination nodes. Our proposed NLBGND optimization algorithm has been implemented alongside with other representatives of well-known algorithms in solving our formulated IoT-FinTech routing problem. Results shown that our algorithm could fairly outperform on many occasions others in finding the optimal solution. This has proven the worthiness of the proposed changes to the native GND algorithm.

For future work, we are planning to consider more descriptive modelling in formulating more data-driven optimization method for IoT related problems. We are also planning to apply the proposed NLBGND algorithm to a set of IoT-related applications such as Industrial IoT (IIoT), Agricultural IoT (AIoT) and Internet of Autonomous Vehicles (IoAV).

REFERENCES

- [1] K. Ashton, "That 'Internet of Things' Thing-RFID Journal," *IoT*, url: <http://www.rfidjournal.com/articles/view>, vol. 4986.
- [2] S. Lohr, "The origins of 'Big Data': An etymological detective story," *New York Times*, vol. 1, no. 1, 2013.
- [3] Y. R. B. Al-Mayouf *et al.*, "Real-time intersection-based segment aware routing algorithm for urban vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 7, pp. 2125-2141, 2018.
- [4] W. Saeed, Z. Ahmad, A. I. Jehangiri, N. Mohamed, A. I. Umar, and J. Ahmad, "A fault tolerant data management scheme for healthcare Internet of Things in fog computing," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 15, no. 1, pp. 35-57, 2021.
- [5] M. Maiti and U. Ghosh, "Next generation Internet of Things in fintech ecosystem," *IEEE Internet of Things Journal*, 2021.
- [6] A. Shin, I. Ryoo, and S. Kim, "Self-organization Scheme of WSNs with Mobile Sensors and Mobile Multiple Sinks for Big Data Computing," *KSII Transactions on Internet & Information Systems*, vol. 14, no. 3, 2020.
- [7] T. Winter *et al.*, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," *rfc*, vol. 6550, pp. 1-157, 2012.
- [8] F. Al-Turjman, "Smart-city medium access for smart mobility applications in Internet of Things," *Transactions on Emerging Telecommunications Technologies*, p. e3723, 2019.
- [9] V. N. Medeiros, B. Silvestre, and V. C. Borges, "Multi-objective routing aware of mixed IoT traffic for low-cost wireless Backhalls," *Journal of Internet Services and Applications*, vol. 10, no. 1, pp. 1-18, 2019.
- [10] A. Musaddiq, Y. B. Zikria, and S. W. Kim, "Routing protocol for Low-Power and Lossy Networks for heterogeneous traffic network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, pp. 1-23, 2020.
- [11] J. Kang, "Mobile payment in Fintech environment: trends, security challenges, and services," *Human-centric Computing and Information sciences*, vol. 8, no. 1, pp. 1-16, 2018.
- [12] H. Stewart and J. Jürjens, "Data security and consumer trust in FinTech innovation in Germany," *Information & Computer Security*, 2018.
- [13] L. Manman, Q. Xin, P. Goswami, A. Mukherjee, and L. Yang, "Energy-Efficient Dynamic Clustering for IoT Applications: A Neural Network Approach," in *2020 IEEE Eighth International Conference on Communications and Networking (ComNet)*, 2020: IEEE, pp. 1-7.
- [14] P. Goswami, A. Mukherjee, M. Maiti, S. K. S. Tyagi, and L. Yang, "A Neural Network Based Optimal Resource Allocation Method for Secure IIoT Network," *IEEE Internet of Things Journal*, 2021.
- [15] J. Manyika *et al.*, "The Internet of Things: Mapping the value beyond the hype," 2015.
- [16] H. Kharrufa, H. A. Al-Kashoash, and A. H. Kemp, "RPL-based routing protocols in IoT applications: A Review," *IEEE Sensors Journal*, vol. 19, no. 15, pp. 5952-5967, 2019.
- [17] S. Mirjalili, J. S. Dong, and A. Lewis, "Ant colony optimizer: theory, literature review, and application in AUV path planning," *Nature-Inspired Optimizers*, pp. 7-21, 2020.
- [18] M. M. Alqarni, A. Cherif, and E. Alkayal, "A Survey of Computational Offloading in Cloud/Edge-based Architectures: Strategies, Optimization Models and Challenges," *KSII Transactions on Internet & Information Systems*, vol. 15, no. 3, 2021.
- [19] Z. Kong, D. Wu, X. Jin, S. Cen, and F. Dong, "Improved AP Deployment Optimization Scheme Based on Multi-objective Particle Swarm Optimization Algorithm," *KSII Transactions on Internet & Information Systems*, vol. 15, no. 4, 2021.
- [20] H. Kharrufa, H. Al-Kashoash, and A. H. Kemp, "A game theoretic optimization of RPL for mobile Internet of Things applications," *IEEE Sensors Journal*, vol. 18, no. 6, pp. 2520-2530, 2018.
- [21] M. Kandori and T. Ui, "Introduction to fundamental issues in game theory and market design," *The Japanese Economic Review*, vol. 71, no. 1, pp. 1-5, 2020.
- [22] X. Song, L. Dong, X. Huang, L. Qin, and X. Han, "Energy-efficient Power Allocation based on worst-case performance optimization under channel uncertainties," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 14, no. 11, pp. 4595-4610, 2020.
- [23] G. Zhao, D. Liu, and J. Wang, "The Game Selection Model for the Payoff Strategy Optimization of Mobile CrowdSensing Task," *KSII Transactions on Internet & Information Systems*, vol. 15, no. 4, 2021.
- [24] H. Fotouhi, D. Moreira, and M. Alves, "mRPL: Boosting mobility in the Internet of Things," *Ad Hoc Networks*, vol. 26, pp. 17-35, 2015.
- [25] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, "Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 964-979, 2016.
- [26] C. Ma, J.-P. Sheu, and C.-X. Hsu, "A game theory based congestion control protocol for wireless personal area networks," *Journal of Sensors*, vol. 2016, 2016.
- [27] H. A. Al-Kashoash, H. M. Amer, L. Mihaylova, and A. H. Kemp, "Optimization-based hybrid congestion alleviation for 6LoWPAN networks," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2070-2081, 2017.
- [28] Y. Akbari and S. Tabatabaei, "A New Method to Find a High Reliable Route in IoT by Using Reinforcement Learning and Fuzzy Logic," *Wireless Personal Communications*, pp. 1-17, 2020.

- [29] A. I. A. Ahmed, S. Khan, A. Gani, S. H. Ab Hamid, and M. Guizani, "Entropy-based fuzzy AHP model for trustworthy service provider selection in Internet of Things," in *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*, 2018: IEEE, pp. 606-613.
- [30] H. Aznaoui, S. Raghay, Y. Ouakrim, and L. Aziz, "A Heuristic Algorithm of Cooperative Agents Communication for Enhanced GAF Routing Protocol in WSNs," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.
- [31] Y. Zhang, Z. Jin, and S. Mirjalili, "Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models," *Energy Conversion and Management*, vol. 224, p. 113301, 2020.
- [32] M. Alaslani, F. Nawab, and B. Shihada, "Blockchain in IoT systems: End-to-end delay evaluation," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8332-8344, 2019.
- [33] K. Niven, D. Holman, and P. Totterdell, "How to win friendship and trust by influencing people's feelings: An investigation of interpersonal affect regulation and the quality of relationships," *Human Relations*, vol. 65, no. 6, pp. 777-805, 2012.
- [34] A. S. Sadiq, S. Khan, K. Z. Ghafoor, M. Guizani, and S. Mirjalili, "Transmission power adaption scheme for improving IoV awareness exploiting: evaluation weighted matrix based on piggybacked information," *Computer Networks*, vol. 137, pp. 147-159, 2018.



Ali Safaa Sadiq is a senior IEEE member and currently a faculty member and course coordinator of Artificial Intelligence and Robotics and a member at Wolverhampton Cyber Research Institute (WCRI) at Faculty of Science and Engineering, School of Mathematics and Computer Science, University of Wolverhampton, UK; he is also an adjunct staff at Monash University and adjunct associate professor at the Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia. Ali has served as a lecturer at the School of Information Technology, Monash University, Malaysia. Previously he has also served as a senior lecturer at the Department of Computer Systems & Networking Department, Faculty of Computer Systems & Software Engineering, University Malaysia Pahang, Malaysia. Ali has completed his first degree in Computer Science in 2004, after that Ali had 5 years of industrial experience in Computer Science and Networking. Ali had his MSc and PhD degrees in Computer Science in 2011 and 2014, respectively. Ali has been awarded the Pro-Chancellor Academic Award as the best student in his batch for both master's and PhD. He has published several scientific/research papers in well-known international journals and conferences. He was involved in performing 5 research grants projects, whereby 3 of them are around network and security and the others in analysing and forecasting floods in Malaysia. Recently he has involved as a co-investigator with a research project CYBERMIND that was funded £91k by Innovate UK Cyber Academic Start-up Accelerator 2020, and currently leading a research project under the same program called TrustMe. He has supervised more than 5 PhD students and 6 Masters students as well as some other undergraduate final year projects. His current research interests including Wireless Communications, Network security and AI applications in networking.



Amin Abdollahi Dehkordi is a scholar in the Department of Computer Engineering, Islamic Azad University, Iran. Amin has completed his masters in the field of artificial intelligence and currently he is a faculty member in the

department of computer engineering. His research interest and skills including computer engineering skills and expertise, Image Processing, Pattern Recognition, Artificial Neural Networks, Swarm Intelligence, and Pattern Recognition Algorithms.



Seyedali Mirjalili is the director of the Centre for Artificial Intelligence Research and Optimization at Torrens University Australia. He is internationally recognized for his advances in Swarm Intelligence and Optimization, including the first set of algorithms from a synthetic intelligence standpoint - a radical departure from how natural systems are typically understood - and a systematic design framework to reliably benchmark, evaluate, and propose computationally cheap robust optimization algorithms. Ali has published over 200 publications with over 25,000 citations and an H-index of 58. As the most cited researcher in Robust Optimization, he is in the list of 1% highly-cited researchers and named as one of the most influential researchers in the world by Web of Science. He is a senior member of IEEE and an associate editor of several journals including Neurocomputing, Applied Soft Computing, Advances in Engineering Software, Applied Intelligence, and IEEE Access. His research interests include Robust Optimization, Engineering Optimization, Multi-objective Optimization, Swarm Intelligence, Evolutionary Algorithms, and Artificial Neural Networks. He is working on the application of multi-objective and robust meta-heuristic optimization techniques as well.



Jingwei Too. Jingwei received the bachelor's degree in mechatronics engineering, in 2017, and the Ph.D. degree in electrical and computer engineering from Universiti Teknikal Malaysia Melaka. His research interests include signal processing, meta-heuristics, machine learning, and data mining. (Based on document published on 18 January 2021).



Prashant Pillai. Prof Prashant Pillai is an award-winning academic leader and visionary who has over 18 years of UK Higher Education experience. He is a globally renowned scholar and outstanding university educator who provides academic direction and leadership to research, business engagement and education within the subject area of computer science at the University of Wolverhampton. He is currently a Professor of cyber security and Director of the Wolverhampton Cyber Research Institute. The Institute comprises of over 10 academic staff and focuses of research in the area of Cyber Security and intelligent and secure cyber physical systems/IoT. Prof Pillai is also the Head of School of Mathematics and Computer Science at the University of Wolverhampton. The school comprises of nearly 50 academic member of staff and 10 research staff.